|| }

1. **PARTE I**

| **1. Antecedentes Personales** |
| --- |
| A continuación, se presenta una tabla en la que debes completar la información solicitada. |

| Nombre estudiante | **Patricia Piñones y Gabriel Serrano** |
| --- | --- |
| Rut | **21.283.155-7 y 20.055.042-0** |
| Carrera | **Ingeniería en informática** |
| Sede | **Padre Alonso de Ovalle** |

| **2. Descripción Proyecto APT** |
| --- |
| En la descripción debes señalar brevemente el nombre de tu proyecto APT y las competencias del perfil de egreso que vas a poner en práctica. Si en tu carrera están definidas las áreas de desempeño, también menciona a qué áreas de desempeño está vinculado el proyecto. |

| Nombre del proyecto | **Clawboard** |
| --- | --- |
| Área (s) de desempeño(s) | Desarrollo de software y arquitectura MVC - Bases de datos - Servicios y APIs - Gestión de proyectos - Experiencia de usuario |
| Competencias | Diseño e implementación de sistema web con metodología de desarrollo de software - Gestión de usuarios, roles y los respectivos permisos para cada rol - Asegurar integridad, disponibilidad y confiabilidad de la información - Aplicar buenas prácticas de trabajo en equipo para desarrollar los proyectos tecnológicos. |

| **3. Fundamentación Proyecto APT** |
| --- |
| A continuación, se presentan distintos campos que debes completar con la información solicitada. Esta sección busca que describas en detalle tu proyecto y justifiques su relevancia y pertinencia. |

| Relevancia del proyecto APT | Hoy los estudiantes de educación superior usan múltiples apps desconectadas (calendario, notas, archivos, recordatorios, pomodoro, flashcards). Esto fragmenta la gestión del estudio, provoca pérdida de información y dificulta el seguimiento de tareas y ponderaciones. Nuestro proyecto propone una plataforma unificada, centrada en el estudiante, que integra gestión académica y productividad en un solo lugar.  En Duoc UC, sede Padre Alonso de Ovalle, enfocado en los estudiantes de primer a cuarto año en apoyo al centro académico, se consolidan las herramientas clave para su desarrollo (asignaturas, calendario, calificaciones, tareas, archivos, pomodoro, flashcards) y notificaciones por correo, mejorando organización, adherencia a plazos y rendimiento. Para el campo de Ingeniería en Informática, el desarrollo integra competencias reales de arquitectura |
| --- | --- |
| Descripción del Proyecto APT | El objetivo principal es crear un espacio ideal para cada estudiante que pueda desarrollar y organizar cada una de sus tareas, apuntes y archivos para lo que es su apoyo académico, desarrollando lo siguiente:   * Autenticación/registro y autorización por roles (admin/estudiante). * Administración de usuarios (crear, editar, activar/desactivar). * Gestión de asignaturas (alta/baja, asociación alumno–asignatura). * Calificaciones y ponderaciones (reglas y validaciones de negocio). * Tareas, exámenes y horarios, integrados al calendario personal. * Papelera (restauración de elementos) y tutorial/ayuda. |
| Pertinencia del proyecto con el perfil de egreso | El desarrollo de ClawBoard exige levantar requerimientos, priorizarlos como historias de usuario y modelar datos relacionales para un dominio académico (Usuario, Asignatura, Evento/Calendario, Calificación). A nivel de ingeniería de software, demanda diseñar una arquitectura MVC por capas (controladores, servicios, repositorios y DTOs), programar el back-end en .NET 8 (C#), aplicar control de acceso con autenticación y autorización por roles, y diseñar/documentar APIs REST que serán consumidas por otros módulos. Para asegurar calidad, incorpora pruebas unitarias y funcionales, manejo de errores y logging básico, además de documentación técnica (manual de instalación/uso de dev y contratos de API). En gestión, se trabaja con sprints, Definition of Done y control de versiones con ramas/PRs y revisión cruzada. Todo lo anterior se alinea directamente con el perfil de egreso de Ingeniería en Informática: análisis, diseño, desarrollo e integración de software; administración de bases de datos; aseguramiento de calidad; y trabajo colaborativo bajo un marco ágil. |
| Relación con los intereses profesionales | Este módulo se alinea directamente con mis intereses en desarrollo web/back-end usando .NET 8 (C#) y patrón MVC, así como en ciberseguridad aplicada a nivel de autenticación y autorización por roles (Identity y políticas). Refuerza mi trabajo con bases de datos relacionales (diseño ER, consultas y reglas de integridad en SQL Server/SSMS) y mi enfoque en gestión ágil de proyectos (sprints, Definition of Done, control de versiones con ramas/PRs y revisión cruzada). |
| Factibilidad de desarrollo del Proyecto APT | El desarrollo del proyecto es posible considerando el tiempo que disponemos y su grado de dificultad para desarrollarlo. **Duración:** semestre académico (≈18 semanas).  **Materiales:** PC del laboratorio y personales, Visual Studio .NET 8, SQL Server, repositorio GitHub.  **Facilitadores:** equipo de 2 personas dedicado al módulo ClawBoard; alcance acotado a prototipo funcional; definición temprana de modelo de datos y contratos de API; planificación por sprints con criterios de aceptación (DoD) y revisión periódica de avances.  **Dificultades:** integración entre módulos y configuración/limitaciones del envío de correos  **Mitigación:** contratos de API y modelo compartido desde el inicio; ambiente de prueba para correos; sprints de integración y pruebas continuas. |

1. **PARTE II**

| **4. Objetivos** |
| --- |
| En este apartado debes definir objetivos generales y específicos del Proyecto APT. Es importante aclarar que los objetivos se deben plantear en forma clara, concisa y sin dar mayores explicaciones, es decir, deben entenderse por sí solos. Se sugiere redactarlos utilizando un verbo en infinitivo, pues ello obliga a precisar acciones concretas. |

| Objetivo general | Diseñar e implementar el núcleo administrativo ClawBoard que gestione usuarios y roles, asignaturas, calificaciones/ponderaciones y eventos académicos (tareas, exámenes, horarios) integrados en un calendario personal, incluyendo papelera y ayuda, y exponiendo APIs estables con pruebas y documentación. |
| --- | --- |
| Objetivos específicos | 1.Análisis de requerimientos para proyecto  2.Implementar autenticación y autorización por roles (admin/estudiante).  2.Desarrollar CRUD de usuarios (crear, editar, activar/desactivar).  3.Implementar gestión de asignaturas y su asociación alumno–asignatura.  4.Modelar e implementar calificaciones y ponderaciones con validaciones de negocio.  5.Construir tareas, exámenes y horarios integrados en el calendario personal (filtros y estados).  6.implementar papelera para restauración controlada de elementos.  7.Publicar contratos de API (Usuario, Asignatura, Evento, Calificación) con ejemplos de uso.  8.Testing, probar las funciones del sitio web con el fin de corroborar su correcto funcionamiento  9.Documentar, asegurar y garantizar las funciones del proyecto |

| **5. Metodología** |
| --- |
| En el siguiente apartado deberás describir la metodología, propia de tu disciplina, que utilizarás para resolver el proyecto APT antes descrito, incluyendo las etapas y métodos de trabajo. |

| Descripción de la Metodología |
| --- |
| Se trabajará con un proceso iterativo-incremental usando prácticas de Scrum (sprints de 2 semanas), con Definition of Done (DoD), control de versiones en GitHub y revisiones formales al cierre de cada sprint. El desarrollo se basa en arquitectura MVC con .NET 8 (C#) y SQL Server.  **Fases principales.**   1. **Requerimientos y priorización:** historias de usuario, criterios de aceptación y reglas de negocio. 2. **Diseño técnico:** arquitectura por capas (controladores-servicios-repositorios-DTOs), **ER** y contratos de API. 3. **Implementación incremental:** Autenticación/roles → Usuarios → Asignaturas → Calificaciones → Eventos/Calendario → Papelera/Ayuda. 4. **Pruebas y calidad:** unitarias (servicios/repositorios), funcionales (flujos clave), integración interna, manejo de errores y logging. 5. **Documentación y transferencia:** manual técnico, contratos de API (Postman/Swagger) y video demo.   **Artefactos.** Backlog priorizado, diagramas **MVC/ER**, código con ramas y PRs, suite de pruebas con evidencias, manual técnico y contratos de API.  **Criterios de aceptación (DoD).** Historia desarrollada, probada (unitaria/funcional), documentada y visible con datos de prueba; sin errores 5xx en flujos principales; endpoints documentados y verificados.  **Recursos.** **Visual Studio .NET 8**, **SQL Server/SSMS**, **GitHub**, **Postman**; **SMTP de prueba** si se valida envío de recordatorios.   * Registros de actividad y manejo de errores implementados. * Endpoints documentados y probados (colección Postman o Swagger). |

| **6. Evidencias** |
| --- |
| A continuación, describe qué evidencias serán evaluadas en el informe de avance y en el informe final de tu proyecto APT. Estas evidencias deben ser acordadas con tu docente. Se entenderá por evidencia los productos que se desarrollen durante el proyecto y cuyo propósito sea visibilizar o documentar cómo se ha implementado el trabajo. |

| **Tipo de evidencia**  **(avance o final)** | **Nombre de la evidencia** | **Descripción** | **Justificación** |
| --- | --- | --- | --- |
| Avance | Requerimientos y Alcance | Backlog priorizado de historias del módulo, con criterios de aceptación y reglas de negocio (usuarios, asignaturas, calificaciones, calendario). | Asegura trazabilidad, define el MVP y alinea expectativas técnicas. |
| Avance | Arquitectura MVC + ER | Diagrama de arquitectura por capas (controladores, servicios, repositorios, DTOs) y diagrama ER (Usuario, Rol, Asignatura, AlumnoAsignatura, Evento, Calificación). | Garantiza coherencia técnica y base de datos consistente. |
| Avance | Seguridad y Roles | Flujo de autenticación/registro y autorización por roles; validaciones server-side básicas. | Habilita el resto de funcionalidades bajo control de acceso. |
| Avance | Gestión Académica Base | CRUD de Usuarios y Asignaturas operativos con datos semilla y casos de borde. | Demuestra funcionamiento inicial del núcleo administrativo. |
| Final | Calificaciones y Ponderaciones | Implementación del modelo de notas con reglas de cálculo y validaciones; evidencias de pruebas. | Acredita exactitud en el rendimiento académico registrado. |
| Final | Eventos y Calendario Personal | Creación/edición de tareas, exámenes y horarios integrados en una vista de calendario con filtros. | Muestra organización temporal confiable para el estudiante. |
| Final | Pruebas y Resultados | Plan, casos, ejecución (capturas/logs), métricas de defectos y correcciones; cobertura mínima en servicios críticos. | Evidencia calidad, estabilidad y cumplimiento del DoD. |
| Final | Desarrollo de manual | Guía de instalación/uso en desarrollo (migrations, seeds, configuración) | Favorece transferencia, evaluación y reproducibilidad del módulo. |

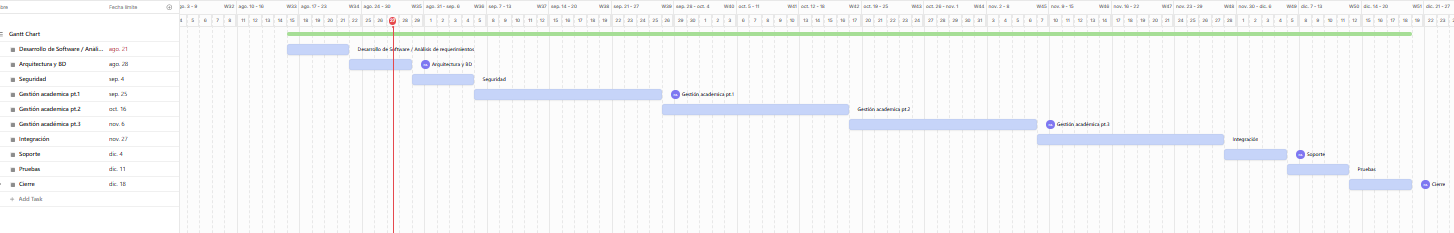
| **7. Plan de Trabajo** |
| --- |
| En la siguiente tabla define la planificación de tu Proyecto APT de acuerdo a lo requerido. |

| **Plan de Trabajo Proyecto APT** | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| Competencia o unidades de competencias | Nombre de Actividades/Tareas | Descripción Actividades/Tareas | Recursos | Duración de la actividad | Responsable[[1]](#footnote-0) | Observaciones |
| Desarrollo de software / Análisis de requerimientos | Levantamiento y backlog | Historias de usuario por módulo, criterios de aceptación y prioridades (MVP). | Plantillas HU, repositorio Git, Docs. | 2 semanas | Gabriel Serrano | Revisar con supervisor |
| Análisis y planificación de sistemas | Definición de requerimientos | Documentación formal de los requerimientos funcionales y no funcionales. Validación con los usuarios. | Plantillas de documentación, UML | 1 semana | Patricia Piñones | Base obligatoria para pasar a la fase de diseño. |
| Arquitectura y BD | Diseñar modelos MVC | Definir la arquitectura mvc para el proyecto | Visual Studio .NET 8, SSMS. | 2 semanas | Patricia Piñones | Revisar dependencias como relaciones |
| Programación backend | Sesión y Usuarios | Desarrollo de autenticación, registro y roles (líder/estudiante). | Visual Studio, C#, Identity API | 2 semanas | Gabriel Serrano | Requiere pruebas de seguridad. |
| Programación backend | Asignaturas | Desarrollo de CRUD de asignaturas para cada estudiante. | Visual Studio, C#, SQL Server | 1 semana | Patricia Piñones | Dependencia del módulo de usuarios. |
| Programaciòn frontend/backend | Crear calificaciones | Implementación del registro, consulta y actualización de notas por asignatura. | Visual Studio, C#, SQL Server | 1 semana | Patricia Piñones | Necesario validar integridad de datos. |
| Programaciòn Frontend | Crear calendario | Creación de calendario académico con horarios, tareas y exámenes. | JS, MVC, librerías de calendario (FullCalendar) | 2 semanas | Gabriel Serrano | Posible dificultad en sincronización con otras funciones. |
| Seguridad | Integración de autenticación y roles asignados | Implementar autenticación con Identity, validaciones serve-side | Visual Studio, SQL Server. | 2 semanas | Patricia Piñones | Probar autenticación con sus permisos. |
| Integraciòn y pruebas | Crear Notificaciones | Implementación de recordatorios vía notificación/correo (AWS SES). | API AWS, Visual Studio | 2 semanas | Gabriel Serrano | Posible retraso si hay problemas con API externa. |
| Pruebas | Unitarias y/o funcionales | Realizar testing a repositorios como servicios en el flujo (desde login hasta la gestión) | Visual Studio, SQL Server | 2 semanas | Gabriel Serrano | Registrar casos en plan de prueba |

| **8. Carta Gantt** |
| --- |
| Busca un formato de Carta Gantt que te acomode y organiza en este las actividades planificadas en el punto anterior considerando el periodo asignado para el desarrollo de tu Proyecto APT. Debes mantener la temporalidad del periodo académico en el desarrollo de las tres fases que contempla la Asignatura de Portafolio de Título. |

| **Actividad** | **Fase 1** | | | | **Fase 2** | | | | | | | | | | | | **Fase 3** | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **S 1** | **S 2** | **S 3** | **S 4** | **S 5** | **S 6** | **S 7** | **S 8** | **S 9** | **S 10** | **S 11** | **S 12** | **S 13** | **S 14** | **S 15** | **S 16** | | **S 17** | **S 18** |
| **Desarrollo de software / Análisis de requerimientos** | **x** | **x** |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| **Definición de requerimientos** |  |  | **x** |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| **Arquitectura y BD** |  |  |  | **x** | **x** |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| **Crear sesiones y usuarios** |  |  |  |  |  | **x** | **x** |  |  |  |  |  |  |  |  |  | |  |  |
| **Creación asignaturas** |  |  |  |  |  |  | **x** |  |  |  |  |  |  |  |  |  | |  |  |
| **Crear Calificaciones** |  |  |  |  |  |  |  | **x** |  |  |  |  |  |  |  |  | |  |  |
| **Crear Calendario** |  |  |  |  |  |  |  | **x** | **x** |  |  |  |  |  |  |  | |  |  |
| **Seguridad** |  |  |  |  |  |  |  |  |  | **x** | **x** |  |  |  |  |  | |  |  |
| **Crear Notificaciones** |  |  |  |  |  |  |  |  |  |  | **x** | **x** |  |  |  |  | |  |  |
| **Pruebas** |  |  |  |  |  |  |  |  |  |  |  |  | **x** | **x** |  |  | |  |  |
| **Presentación** |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** | **x** | | **x** | **x** |

La carta Gantt realizada fue creada desde otra página, el cual se encuentra su link a continuación para poder visualizar de mejor manera: <https://app.clickup.com/90132286940/v/li/901318756607>



1. En caso de que el Proyecto APT sea grupal, en esta columna deben indicar el nombre de los responsables de cada tarea o actividad. Esto posteriormente permitirá diferenciar la evaluación por cada integrante. [↑](#footnote-ref-0)